An Optimal Solution for the Stable Marriage Problem using Particle Swarm Optimization and Firefly Algorithms

Kirti Pandey[1] , C K Jha[2]

[1]*Department of Computer Science, Banasthali Vidyapith, India*

[2]*Department of Computer Science, Banasthali Vidyapith, India*

Abstract:- This paper proposes a hybrid approach to solving the Stable Marriage Problem (SM) by fusing ideas from the firefly optimization (FA) and particle swarm optimization (PSO) methods. It is commonly known that there is always at least one stable matching exists for each instance. The conventional Gale-Shapley formula, on the other hand, produces a partnership that either strongly favours men over women or vice versa. Our proposed approach uses a hybrid optimization technique to solve the problem of stable marriage by providing the best possible pairs. The fitness function is defined here on the basis of the type of optimal solution. To categorize the kind of optimal solution two score functions are defined in this work.The proposed research uses FA to explore the undiscovered solution space, in contrast to our suggested technique which is based on PSO. The particles are then subjected to a dynamic adaptation mechanism for certain parameters, which gradually raises their values as part of the process of determining the velocities. The generation number of the swarm is the dynamic value for these parameters. These two parameters will use their predetermined values for the remaining particles in the swarm. It's worth noting that the dynamic values will be used by different particles in each generation. The proposed method is found to be superior than the conventionaltechniques in terms of the performance scores representing the quality of matching for different types of optimal solutions.

Keywords*:* *Combinatorial problems, firefly optimization, particle swarm optimization,stable matching/marriage, dynamic adaptation mechanism..*

1. Introduction

The difficulty of pairing agents from two sets of size n in such a way that each agent maintains a preference ranking over those in the other set is referred to as the stable marriage problem (SMP). This challenge appears in every market with two distinct buyers and sellers, such as when pairing up medical experts with healthcare facilities, students with educational institutions, or sailors with vessels [1-3]. The difficulty lies in selecting suitable companions for each individual man and woman so that they do not wind up with somebody that they would rather not be with. According to the conventional interpretation, the two categories are composed of potential husbands and wives. We refer to the pairing as a whole as stable when there are no instances of such a pair being present. Gale and Shapley [4] were the first to pose this problem, and they proved that a method with a constant-time complexity of O(n2) can always identify a stable answer. Fortuitously, the Gale-Shapley Stable Marriage Algorithm (SMA) can only provide a male optimal or a female optimal solution [5], which means that it can only produce a solution in which the males are extremely happy while the females are extremely unhappy. As a result, it is not suitable for the overwhelming majority of real applications, the bulk of which require robust and fair matchings. As a direct consequence of this, the equitable stable marriage issue (ESMP) has been the subject of its very own body of study [6-9]. Finding a stable equilibrium that is fair to all sides of the market that is being studied is a requirement of the ESMP that you must fulfil. To be fair, it is necessary to ensure that neither party is given the impression that they have been treated unfairly. However, achieving greater fairness shouldn't come at the expense of people's ability to have fun in general because the two goals don't always go hand in hand. It is not desirable to find a solution that would leave both men and women feeling equally unsatisfied with the outcome of the situation.

Unfortunately, maximizing equity as the only objective is an NP-hard problem [10]. Approximation algorithms [11] and heuristics [12–15] have been suggested as potential solutions to this problem in more recent research. However, as of right now, no practicable approach that can competently produce reasonable and stable matches for key issue scenarios. The usefulness of a local search algorithm and the practicability of providing an FPT technique or FPT approximation is evaluated in [16]. This evaluation may be found in the reference. Finding a long-term companion by means that are egalitarian or that cause the least amount of regret. The parameterized complexity paradigm is applied to a variant of the traditional stable marriage problem known as the hospitals/residents with couples' problem in [17,18]. The residents and hospitals problem has been extended here so that resident teams of two can now submit rankings to hospitals. The authors make use of a local search technique and study the practicability of building FPT algorithms that are able to operate in this setting.

In the paper [19], Gent and Prosser use constraint programming to encapsulate the stable marriage problem with ties and incomplete lists (SMTI). They then give a comprehensive empirical investigation of the issue. After that, the encoded problem can be solved by applying the currently available CPtechnology. Both the choice problem, "Is there a stable matching of size n?" and the optimization problem, "Where can we find a stable matching with the highest or lowest cardinality?" have potential answers that we have proposed. Their all-encompassing approach identifies stable marriages with an average size of 9.3 for situations with a size of 10 and no ties. This is based on solving the CP encoding of the problem with the Choco constraint programming toolkit [20]. The size of the knot grows proportionally with the number of ties used. A distributed solution to the same problem of SMTI is discussed in [21]. This method protects the users' preferences while resolving the issue of stable marriages. They adapt a modified version of Gale Shapley (G-S) algorithm, so that they can function well in a decentralized environment. In addition to this, we provide a general strategy for the resolution of issues involving distributed constraint programming. They take into account the costs of computing (in the form of constraint checks), as well as the effort required for communication; however, neither of these factors is significant to our centralized strategy. In addition to this, the number of unions that their algorithms found when working with SMTI data is displayed.

Two heuristic strategies are presented in [22] to discover the largest SMP which are built on different iterations of the G-S algorithm, with the first system catering to residents and the second system focusing on medical facilities. Heuristics are utilized in order to determine the best way to bring an end to a relationship in order to maximize the size of the subsequent marriage. In actual practice, the methods that are used to break ties can have a large impact on the total size of the stable matching that is revealed. The sizes of two matches can potentially differ by a factor of two in the worst possible scenario. The problem transforms into an SMTI instance with ties on just one side when the capacity of hospitals is set to 1, and this allows the solutions given in [22] to be applied to it.

Maximum cardinality matching (MCM) has been found in [23] that allows the fewest blocking pairs in a SMP instance and offer results on the difficulty and approximation of this work. They do this by presenting a solution to the problem of finding a MCM that allows the fewest blocking pairs. They provide evidence that demonstrates how NP-hard this problem is. The findings of our experiments indicate that using a local search strategy allows us to locate marriages with large numbers of participants and a low number of objectionable spouses in a relatively short amount of time.

Surafel and Hong [24] improved FA so that it could continue looking for the best possible answer for an infinite amount of time by modifying the intensity of the firefly as it flits around in a random pattern. Gandomi et al. [25] claim that merging FA with chaotic maps can boost the convergence performance of traditional FA by increasing the available space for global search. This is accomplished by increasing the amount of information that can be gathered from the map. Within the framework of the recommended strategy, the rule of adaptation for the exploration parameter was outlined with reference to the light absorption coefficient. Kazem, et al. [26] developed a stock market forecasting method by combining the capabilities of support vector regression (SVR) with the advantages of chaotic factor analysis. The best possible settings for the SVR hyper-parameters were determined with the help of chaotic FA. In order to overcome a difficulty associated with combinatorial optimization, Falcon et al. [27] developed a Binary FA that is predicated on binary encoding. They claimed that in contrast to other nature-inspired algorithms, their approach to the candidate solutions avoided the unrealistic responses to the problem and only displayed the feasible ones. This was in contrast to previous algorithms that were inspired by nature. In order to deliver the very best possible CVRP service, Goel and Maini [28] combined the ACO and FA algorithms in order to provide a solution to this challenge. They utilized FA to investigate the possibilities, and then they bolstered it with an ACO feature known as the pheromone shaking approach to protect themselves from entering a situation in which they would be trapped in a local minimum. When compared to the conventional optimal processes, the proposed method is thought to perform better in terms of producing the best answers after the same number of iterations as those processes. Saghaeeian and Ramezanian [29] integrated GA and FA algorithms for use in a duopolistic market and then applied a business solution strategy to the situation. The authors constructed a GA with the use of a technique called local search priority encoding. Following that, a multi-stage crossover strategy is utilized in order to carry out the metaheuristicoptimization.

In the course of this investigation, PSO was combined with the multi-modal capabilities of FA in order to extend the scope of the search and come up with a hybrid solution to the SMP problem. Because FA is designed to address challenges of continuous optimization, it necessitates the careful tuning of a number of factors, such as population initialization and the selection of neighbourhoods, among others. As a consequence of this, we offer an innovative encoding approach for the purpose of depicting firefly populations as well as local locations. A novel hybrid model that incorporates parts of the FA and PSO approaches has been developed as the primary contribution of this body of work. The objective of this model is to locate the best possible solution to a combinatorial problem of stable marriage/matching. The framework of the approach that we have suggested is based on PSO, but we make use of FA in order to narrow down the space of potential solutions that has not yet been investigated. PSO has implemented the pheromone shaking approach in order to prevent pheromone stagnation on exploited areas. This is done in order to further avoid becoming trapped in the local optimal state. In this study, we alter the FA framework by adding a robust stopping criterion in order to attain the greatest possible performance in terms of the amount of computational complexity, the amount of time it takes, and the amount of noise it suppresses.

The remaining parts of the paper are aranged in the following way: The mathematics that underpin SMP and the Gale-Shapley approach are discussed in Section 2, which offers a condensed introduction to the subject. In this third and last part of the series, we will review PSO and FA, the two algorithms that are currently being utilized. The improvements that need to be done as well as the method that is proposed are discussed in Section 4. In Section 5, we undertake an SMP problem analysis utilizing the hybrid model that was recommended, and we evaluate the model's overall performance. In the last part of this investigation, we will discuss the things that we have discovered as a result of our work.

2. Problem formulation

a) Stable Matching/Marriage Problem (SMP)

In order for a system to finish an SMP, it must compare and match two different groups of N objects. As a result of the fact that the two sets of elements each contain N men and N women, it is commonly stated in the research that each individual must be accompanied by someone who is of the opposite sex. Both men and women have preference lists that outline the specific members of the other sex's population whom they find attractive. Preference lists can be found in both public and private spaces. The perfect pair would feel secure in their relationship to the point where neither one of them would choose to marry anybody other than their existing companion.Many different types of experts, including mathematicians, economists, game theorists, computer scientists, physicists, and others, have been interested in this issue. It was first brought to light in 1962 by the key work that was done by Gale and Shapley. Most importantly for our purposes, SMP can be viewed as a combinatorial optimization problem, in which the goal is not to find a solution that is stable but rather to find a solution that maximizes global pleasure (the higher your spouse ranks on your list of preferences, the happier you will be personally). The terms "situational matching problem" and "random matching problem" are synonymous with one another.

An instance of the SMP consists of n men and n women, and both sexes provide a list of the other sex's members in linear order. This list is then compared and ranked by the number of stable marriages that result from the problem. A preference list is one type of list that falls under this category. If there is only one connection between the male and female populations, then we refer to that as a marriage, which is denoted by M. Assuming that $(m, w) \in M$exists, we can either state that m is married to w inM or that w is married to m in M, or we can say that(m, w) is a pair in M. In addition, we assert that a person in M is free to act in any manner that they want. They are regarded as being matched in M if none of them is married to any other woman (man) in M at the time of the match. In the following, male members of M who are married to female members of M will be abbreviated as M(w), while female members of M who are married to male members of M will be abbreviated as M(m).

Let's consider A, B & C are the person living in such a way that B & C are having same gender and A is having opposite gender. It has been state that A would prefers B to C if B appears in the left side range from C in A's preference list. Let's consider L is the set of male and M is set of females and let assume L' belongs to L x M be a marriage. So, pair $(a, b) \in L \times M$is a blocking pair for L' if at least one of the following conditions is satisfied:

1. Both a & b is free in L'
2. a is free in L' and b prefers a to L'(b)
3. b is free in L' and a prefers w to L'(a)

L' is considered as a stable marriage if there is no blocking pair for L'.

b) Gale–Shapley (GS) Algorithm

When N is large enough, it becomes difficult to find stable solutions (note that there may be several). In 1962 [4], Gale and Shapley presented the first and most well-known approach for finding a particular stable solution. The algorithm goes by the name Gale-Shapley (G-S).There are separate iterations of the GS algorithm tailored specifically to males and women. Since the mechanics are consistent throughout iterations, we will only highlight the one that is designed for males.After receiving a problem instance, the algorithm executes the following steps:

1. First, work towards a world in which gender equality is universally achieved.
2. m, a man without ties, has written the following proposition to the most attractive woman on his list, whom he has yet to approach.
3. mi proposes to wj because she is the most desirable woman on his list of potential partners.
4. If and only if wjgets some downtime, mi and wj will get engaged.
5. If mi has feelings for mp (i.e., yj,pyj,i), wj will turn down her proposal, leaving him unmarried.
6. If wj feels mp is better than mi(yj,p>yj,i), she will break off her engagement with mp and start dating mi. Therefore, there is no penalty for mp's return.

This step is repeated until every single man has found a wife.Similarly, in the female-centric version, women are the ones who do the proposing. It is likely to prove that the algorithm finds a stable solution and that the solution found in the men-oriented version is the best stable solution that can be obtained for men. This means that out of all the possible mates a man could have in the other stable solutions, he will get the best companion in this one. In this case, it's believed that men provide the optimal solution. In a similar vein, the version that places an emphasis on women reaches a solution that is best for women in the long run.The ideal solution for males is also the worst solution for women, and vice versa for the optimal solution for women. In the end, it's the person who takes the initiative who benefits.

The runtime complexity of this method is O(n2), where n is the total number of people. The GS algorithm's pseudocode is presented in the following algorithm.

Algorithm 1: Gale Shapley Algorithm

Initialize all *m* and *w* to free

while there is a *m* who has non-empty list **do**

  w=*top* in the list

  if *w* is free **then**

    mw engage

  else

    some *m′w* already engaged

    if *w* prefers *m* to *m′* **then**

      mw engage, *m′* becomes free

      Remove *w* from *m′*'s list

    else

      m′w remain engaged, remove *w* from *m′*'s list

  end if

end while

I. Brief Overview Of Firefly Algorithm And Pso

a) Firefly Algorithm

The firefly technique provides an innovative population-based metaheuristic that may be applied to continuous FA problems. The ambition to imitate the social behaviour of flashing lights exhibited by fireflies was the driving force for FA. One way to identify a firefly is by the specific brightness of its light emission.[30]
These are the rules that the firefly algorithm follows:

- All fireflies are attracted to one another, regardless of gender.
- The dimmer fireflies are drawn to the brighter ones, which gives the brighter fireflies the opportunity to advance to the next destination ahead of the others. One way to conceptualize the ultimate objective function is as the degree to which a firefly shines its light.

$$I_x \propto TC \qquad (1)$$

where Cx and Ix represent the cost function and light intensity respetively.

- When there are two fireflies of differing brightnesses, the first firefly will fly towards the direction of the firefly that is brighter, and the solution will be modified to reflect this. The following formula, which takes into account the mutation coefficient (α), attraction coefficient (β), and light absorption coefficient(δ), can be used to depict the occurrence of new position (x') for fireflies.

$$x_i' = x_i + \beta_0 e^{-\delta r^2}\left(x_j - x_i\right) + \alpha\varepsilon \qquad (2)$$

Because of its innate ability to separate into smaller groups, FA is able to categorise fireflies into a broad number of different subgroups. This is the key reason why FA is so effective when dealing with problems that involve both linear and nonlinear components. On the other hand, the position of FA breathes new life into the heading, despite the fact that it is predominantly one-dimensional and diagonal. Because it cannot search with a geographical bias, FA requires a significant amount of processing power and lacks diversity. Many other lines of research have followed suit by adopting a version of the FA framework that has been slightly modified and now includes an encoding mechanism. Figure 1 depicts the fundamental architecture of the FA [31].

b) PSO Algorithm

PSO is inspired by the cooperative behaviour of flocks of birds, and was first proposed by Kennedy and Eberhart [32]. Each i, or particle, is a solution to the optimization problem, which is represented as a vector of choice variables xi.

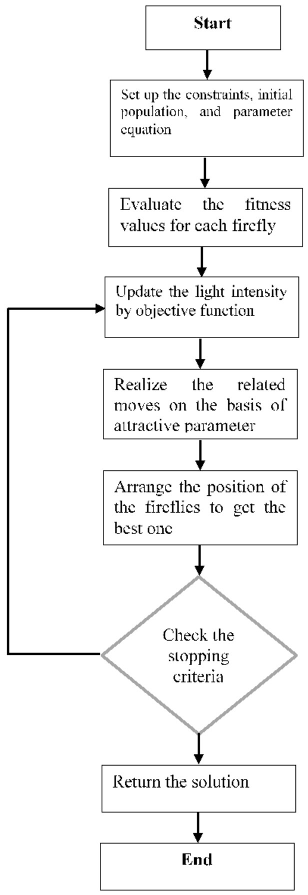

Figure 1. Flowchart of the firefly algorithm [31].

The particle with the highest fitness value acts as the swarm's (population of particles') leader, guiding the swarm to productive areas of the search space. Cognitive (i.e., its current best position, abbreviated xpbesti) and social (i.e., the position of the swarm leader, abbreviated xgBest) information influence the search direction taken by each particle. The swarm's leader evolves with each cycle of the procedure. Depending on the particle's current velocity, vi(t) (search direction), the following two elements are employed to compute the particle's new velocity, vi(t+1):

$$v_i(t+1) = v_i(t) + c_1 r_1 \left(x_{pbest_i} - x_i\right) + c_2 r_2 \left(x_{gbest_i} - x_i\right) \quad (3)$$

where r1, r2 are uniformly distributed random numbers between 0 and 1, while c1and c2 are acceleration coefficients to govern the effect of cognitive and social information, respectively.
The positions of each particle are updated by flight formula after it updates its related velocity:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (4)$$

Where the current position and new position of thisparticle is represented by xi(t)andxi(t +1) respectively, and vi(t +1) is its newly updated velocity.Each particle should include an n, or neighbourhood ideal position, to prevent the swarm from becoming trapped in a local minimum. Here's when the social aspect of particle swarm optimizations comes into play.

The swarm's population topology can be understood as the network of interpersonal relationships. The gbest topology is just one example of the many possible variants.All particles tend to congregate around the globally optimal solution, which in this topology stands in for a fully integrated social network. On the other hand, the lbesttopology simply connects each particle to its C neighbours. This topology has a slower convergence rate than gbest, but it aids PSO algorithms in avoiding local minima. The wheel topology only links one particle to its neighbours, the focus particle. This is the least popular form of connection.

In contrast to other metaheuristics, the PSO algorithm in the STP only requires the computation of two variables per iteration: the location and velocity of each member of the population. Using local search metaheuristics and the path relinking technique, we improve the performance of Particle Swarm Optimization.

3. Proposed Methodology

This work proposes a hybrid of PSO and FA to solve the NP-hard problem of stable matching. PSO use FA search to probe potential outcomes. Additionally, a pheromone exchange procedure is proposed to forestall convergence before its time. Our proposed method utilizes a hybrid algorithm with PSO as its framework and FA for its search space exploration. Random particles representing different solutions are created in the PSO method. Every atom has a fixed position in the solution space and travels at a constant velocity. Each member of the population (a "particle") is represented by a d-dimensional vector in the problem space denoted by $x_i = (x_{i1}, x_{i2},..,x_{id})$, $i = 1, 2,..., N$ (where N is the size of the population and d is the number of dimensions of the vector). Here, the fitness function is categorized as man-optimal, woman-optimal, egalitarian, or sex-fair depending on the nature of the optimal solution. In this work, we define two score functions, man score $sm(M_i)$ and woman score $sw(Mi)$, to classify the type of best solution. A woman's score is based on where she appears on a man's list, and vice versa. Here is a mathematical description of the scores:

$$sm(M_i) = \sum_{(m,w)\in M_i} mr(m,w)$$

$$sw(M_i) = \sum_{(m,w)\in M_i} wr(m,w)$$

where$mr(m,w)$ is the rank of w in the list of m and $wr(m,w)$ is the rank of m in w's list. The definitions of the type of optimal solution are given in table 1:

Table 1. Type of optimal solution

Fitness function	Type of optimal solution
$\max sw(M_i)$ over all $M_i \in M$	Man-optimal
$\max sm(M_i)$ over all $M_i \in M$	Woman-optimal
$\min (sm(M_i) + sw(M_i)$ over all $M_i \in M$	Egalitarian
$\min \lvert(sm(M_i) - sw(M_i)\rvert$ over all $M_i \in M$	Sex-fair

The adjustments that will be made to shift the particle from one point to another are represented by the velocity vij. The dynamic transitions of the particles in the problem of STP are done as per the type of optimal solution and the respective fitness function. For example, if Man-optimal solution is required for the problem of stable marriage, then maximization of $sw(M_i)$ is taken under consideration. The objective function or fitness functions mentioned in table 1 will govern that what type of optimal solution is going to be generated. The dynamic interaction of the particle's individual experience and the experience of the entire swarm determines where it will move. A particle can travel in one of three directions: continue its own route, move to the best position it had during the iterations (pbestij), or move to the best particle's position (gbestj).Clerc and Kennedy [33] developed a constriction coefficient, k, that affects all values involved in the velocity update as follows in order to eliminate velocity clamping and encourage convergence

$$v_i(t+1) = k[v_i(t) + c_1 r_1 (x_{pbes\ t_i} - x_i) + c_2 r_2 (x_{gbes\ t_i} - x_i)] \quad (5)$$

The two most crucial variables in the velocity update formula (Eq. 3) are (1) k, which affects the velocity's overall value, and (2) c2, which has a greater impact on the calculation because, in the majority of cases, the pbest value corresponds to the particle's current position while the gbest value differs from the particle position throughout the search process (aside from the leader). Additionally, PSO has a propensity to solve problems. Then, as illustrated below, it is advised to utilize a dynamic (deterministic) adaptation approach for these two parameters, k and c2, starting with low velocity values for some particles and gradually increasing them as the search proceeds. Some (variable) fraction of the particles in the swarm will utilize a dynamic value for k and c2based on the generation number. The specified values for these two parameters will be used by the remaining swarm particles. It's crucial to keep in mind that different particles in successive generations will utilize dynamic values; for example, a specific particle might use fixed values in generation "t" and dynamic values in generation "t +1". Dynamic values are intended to cause particles to advance more slowly than their counterparts in each generation. It is desirable to have a slower convergence rate in order to get higher performance, which entails more trustworthy and high-quality outputs. A dynamic variation was developed to take advantage of PSO's quick convergence, in which the k and c2 values expanded more quickly for the second half of the process after being kept low for the first half of the process (i.e., half of the total generations).
These solutions were created using the FA search, and the solution pool served as a firefly breeding environment. The pheromone experiment can be enhanced for upcoming runs using the best solution available right now. Algorithm 2 lays up the specifics. The original swarm intelligence systems' flaws in both of them are addressed by the hybrid algorithm that is being proposed. In order to locate unexplored areas, it combines pheromone shaking with FA search.

Algorithm 2
Step 1: Initialization

a) **Select**the number of swarms for the respectivevariant of the PSO algorithm
b) **Select**the number of particles in each swarm
c) **Initialize** the position and velocity of each particle
d) **Transform**the positions of particlesin continuous mode
e) **Choose**the standard deviation ofdemands.
f) **Select**the type of optimal solution for the SMP i.e. man-optimal, woman-optimal, egalitarian or sex-fair.
g) **Initialize** the man-woman pairmatchingat in random manner.
h) **Initialize**the initial fitness function of each particle
i) **Identify**the best solution of each particle
j) **Find** the best particle of the entire swarm

Step 2:PSO Phase

a) **Do while** the maximum number of iterations has not been reached
b) **Calculate** the velocity of each particle
c) **Calculate** the new position of each particle
d) **Convert** particles' positions in integer form
e) **Calculate** the new fitness function of each particle
f) **Improve** the solutions with 2-opt, 3-opt, Path Relinking
g) **Update** the best solution of each particle
h) **Find** the best particle of the whole swarm
i) **Convert** particles' positions in continuous form
j) **Enddo**

Step 3 : FA Search (to search for the other promising regions which are lessexplored)

1. Search Inception.

You may think of the 'm' tours that PSO produces as a population of 'm' fireflies.

2. Calculation of Light Intensity.

The intensity of each firefly's flash is set according to the type of optimal issue and the associated objective function. It is set to be equal to the inverse of the objective function value for the minimization objective function. Additionally, the objective function value is set to be identical to the objective function for maximisation.

3. Movement&Attractiveness

Sort 'm' fireflies based on how bright they are. The best firefly can be thought of as having the lowest objective function value. A function of light intensity is used to calculate the best firefly's attractiveness to other fireflies.

$$\beta = \beta_0 e^{-\gamma r_{ij}^2}$$

where β_0 is the attractiveness at $r = 0$ and r_{ij} is Euclidean distance between two fireflies atpositions i and j. Besides, the movement of fireflies can be controlled as a function of attractiveness (β) andrandom move (α) given by

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} + \alpha$$

Here the second term is the attractiveness and third term α is the randomness control parametersuch that $\alpha \in [0, 1]$.

4. Return "*m*" explored tours to PSO algorithm.

Step 4 :*For Man-optimal and Woman-optimal problem (Maximization)*: Among above solutions updated by FA, select a particle having largest objectivefunction as $best_{\text{part}}$ and set $G_{best} = best_{\text{part}}(len)$.

For Egalitarian and Sex-fair problem (minimization): Among above solutions updated by FA, select a particle having smallest objectivefunction as $best_{\text{part}}$ and set $G_{best} = best_{\text{part}}(len)$.

Step 5. Output $Gbest$ and Stop.

The global cost function and cost measure on couples will vary depending on the stable matchingfinding criterion and type of optimal solution as mentioned earlier. The hybrid PSO-FA model determines the best match based on the objective function.

4. Experimental Results

We ran performance simulations of the suggested algorithm with n (the number of male participants in the challenge) ranging from 3 to 20. The Gale-Shapley algorithm was also simulated, and the outcomes were compared. Findings for both man- and woman-optimal stable matching are identical to those of the Gale-Shapley algorithm. Matlab R2009a was used to implement the entire algorithmic technique. Table 2 shows the chosen values for the parameters taken for the hybrid optimal model of PSO and GA.

Table 2. Parameters for the proposed model

Parameters	Value
Maximum number of particles(m)	80
Maximum Iterations (N_{max})	10
Attractiveness control parameterof FA (γ)	0.5
Randomness control parameter (α)	0.1
Tolerance for equality constraint for	0.0001

Population size	40-120

The results of a comparison of our suggested model with the Gale-Shapley approach, the genetic algorithm(GA), and the Ant Colony Optimisation (ACO)are presented in Table 3.

Table 3. Comparison of proposed model with conventional techniques

Score for the respective type of optimal solution	**Gale-Shapley**	**ACO**	**GA**	Proposed PSO-FA
Man-optimal-$sw(M_i)$	77	77	77	**77**
Woman-optimal-$sm(M_i)$	82	82	82	**82**
Egalitarian-$(sm(M_i) + sw(M_i)$	-	107	103	**100**
Sex-fair-$\|(sm(M_i) - sw(M_i)\|$	-	8	4	**2**

Within the context of the sex-fair discovery experiment, we carried out a total of 90 trials, with n varying anywhere from 6 to 20 (6 for each run). In ninety percent of cases, the difference in score between a man and a woman was lower than three, and in 94 percent of cases, it was lower than four. In the testing where it was identified, we found man-optimal, woman-optimal, and egalitarian stable matching for the same instance. All of these results were for the same person. We determined the greatest possible scores for women *(sw*)*, men *(sm*)*, and egalitarians *(sw+sm)* by calculating the best possible score for each category. The egalitarian score, denoted by the symbol *Se**, was found to be very close to being equal to *2(sw*+sm*)*. The outcome demonstrates whether or not this stable matching is fair or very close fair.

5. Conclusion

To resolve the well-known combinatorial optimization problem of stable marriage/matching, a hybrid PSO-FA method is suggested. PSO and fireflies are hybridized in this study to create variety in the solution space because FA has a multi-modal nature. FA must be applied with special modifications to population initialization, neighbourhood selection, and other aspects because it was created for continuous optimization problems. A new coding technique is consequently suggested for encoding the population pool and neighbourhood representation of fireflies. An innovative firefly representation and distance measuring technique has been given to fit FA for such discrete optimization issues. According to the sort of optimal solution—man-optimal, woman-optimal, egalitarian, or sex-fair—the fitness function is defined below. Two score functions are created in this study to classify the various types of optimal solutions. The effectiveness of the suggested hybrid technique is contrasted with that of a number of other current techniques, including Ant Colony Optimization (ACO) and Genetic Algorithm (GA). The proposed method outperforms earlier methods in terms of performance ratings that reflect the effectiveness of matching for various sorts of optimal solutions.

Refrences

[1] RobertW. Irving and David F. Manlove. Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems. Journal of CombinatorialOptimization, 16(3):279–292, 2008.

[2] Kazuo Iwama, Shuichi Miyazaki, David Manlove, and Yasufumi Morita. Stable marriage with incomplete lists and ties. In ICAL '99: Proceedings of the 26th International Colloquium on Automata, Languages and Programming, pages 443–452,London, UK, 1999. Springer-Verlag.

[3] Kazuo Iwama, Shuichi Miyazaki, and Naoya Yamauchi. A 1.875: approximation algorithm for the stable marriage problem. In SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pages 288–297,Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[4] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," The American Mathematical Monthly, vol. 69, no. 1, pp.9–15, 1962.

[5] Gelain, M.; Pini, M.S.; Rossi, F.; Venable, K.B.; Walsh, T. Local search algorithms on the stable marriage problem: Experimental studies. In Proceedings of ECAI 2010, Lisbon, Portugal, August,2010; pp. 1085–1086.

[6] Gelain, M.; Pini, M.S.; Rossi, F.; Venable, K.B.; Walsh, T. Local search for stable marriage problems with ties and incomplete lists. In Proceedings of PRICAI 2010, LNCS 6230, Daegu,Korea, August 2010; pp. 64–75.

[7] Gelain, M.; Pini, M.S.; Rossi, F.; Venable, K.B.; Walsh, T. Procedural fairness in stable marriageproblems. In Proceedings of AAMAS 2011, Taipei, Taiwan, May 2011; pp. 1209–1210.

[8] Iwama, K.; Miyazaki, S.; Yanagisawa, H. A 25/17-Approximation Algorithm for the StableMarriage Problem with One-Sided Ties. Algorithmica 2012.

[9] Manlove, D.; Irving, R.W. Approximation algorithms for hard variants of the stable marriage andhospitals/residents problems. J. Comb. Optim 2008, 16, 279–292.

[10] McDermid, E., Irving, R.W.: Sex-equal stable matchings: complexity and exact algorithms. Algorithmica, vol. **68, no.** 3, pp. 545–570, 2014.

[11] Kir´aly, Z. Better and Simpler Approximation Algorithms for the Stable Marriage Problem.Algorithmica 2011, 60(1), 3–20.

[12] McDermid, E. A 3/2-approximation algorithm for general stable marriage. In Proceedings ofICALP 2009, LNCS 5555, Rhodes, Greece, July 2009; Volume 1, pp. 689–700.

[13] Manlove, D.F. Algorithmics of Matching Under Preferences; World Scientific Publishing: Singapore, 2013.

[14] Kasperski, A., Zieli´nski, P.: Combinatorial optimization problems with uncertain costs and the OWA criterion. Theor. Comput. Sci., vol. 565, pp.102–112, 2015.

[15] Bertrand Zavidovique, NikomSuvonvorn, Guna Seetharaman, A novel representation and algorithms for (quasi) stable marriages, in: ICINCO,2005, pp. 63–70.

[16] Bir´o, P.; Manlove, D.F.; Mittal, S. Size versus stability in the marriage problem. Theo. Com. Sci.2010, 411, 1828–1841.

[17] Pini, M.S.; Rossi, F.; Venable, K.B.; Walsh, T. Manipulation complexity and gender neutrality in stable marriage procedures. J. Aut. Agents Multi-Agent Sys. 2011, 22, 183–199.

[18] Pini, M.S.; Rossi, F.; Venable, K.B.; Walsh, T. Stability and Optimality in Matching Problems with Weighted Preferences. In Agents and Artificial Intelligence 2011 - ICAART 2011 – RevisedSelected Papers, CCIS 271, Roma, Italy, 2012; pp. 319–333.

[19] Gent, I.P.; Prosser, P. An empirical study of the stable marriage problem with ties and incompletelists. In Proceedings of ECAI 2002, Lyon, France, July 2002; pp. 141–145.

[20] Ioannis Giannakopoulos, et al., An equitable solution to the stable marriage problem, in: 2015 IEEE 27th International Conference on Toolswith Artificial Intelligence (ICTAI), IEEE, 2015, pp. 989–996.

[21] Brito I.; Meseguer, P. Distributed stable matching problems with ties and incomplete lists. In Proceedings of CP 2006, LNCS 4204, Nantes, France, September 2006; pp. 675–679.

[22] Irving, R.W.; Manlove, D.F. Finding large stable matchings. J. Exp. Algorithmics 2009, 14, 1.2:1–1.2:30.

[23] MaximeMorge, Gauthier Picard, Privacy-preserving strategy for negotiating stable, equitable and optimal matchings, in: Advances on PracticalApplications of Agents and Multiagent Systems, Springer, 2011, pp. 97–102.

[24] SurafelLulesegedTilahun and Hong Choon Ong, (2012) "Modified Firefly Algorithm," Journal of Applied Mathematics, Article ID 467631, 2012.

[25] Gandomi, A. H., Yang, X. S., Talatahari, S., &Alavi, A. H., "Communications in Nonlinear Science and Numerical Simulation, vol. 18, no.1, pp. 89-98, 2013.

[26] Kazem, A., Sharifi, E., Hussain, F. K., Saberi, M., & Hussain, O. K., "Support vector regression with chaos-based firefly algorithm for stock market price forecasting", Applied soft computing, vol. 13, no.2, pp. 947-958, 2013.

[27] Falcon, Rafael, Marcio Almeida, and Amiya Nayak. "Fault identification with binary adaptive fireflies in parallel and distributed systems." IEEE Congress on Evolutionary Computation (CEC), 2011.

[28] Goel, Rajeev, and Raman Maini. "Vehicle routing problem and its solution methodologies: a survey." International Journal of Logistics Systems and Management, vol. 28, no. 4, pp. 419-435, 2017.

[29] Amin Saghaeeian, Reza Ramezanian, "An efficient hybrid genetic algorithm for multi-product competitive supply chain network design with price-dependent demand", Appl. Soft Comput. Vol. 71, pp. 872-893, 2018.

[30] Yang, Xin-She., "Firefly algorithm, stochastic test functions and design optimization." International Journal of Bio-Inspired Computation, vol. 2, no.2, pp. 78-84, 2010.

[31] Farahani, S. M., Abshouri, A. A., Nasiri, B., &Meybodi, M., "A Gaussian firefly algorithm", International Journal of Machine Learning and Computing, vol. 1, no. 5, pp. 448-462, 2011.

[32] Kennedy, J., Eberhart, R., "Particle swarm optimization", In: Proceedings of the IEEE International Conference on Neural Networks, pp. 1942–1948, 1995.